ScarCoin Mainnet Deployment Strategy

Security-First Approach to Production Launch

Version 1.0 | August 30, 2025

Comprehensive Deployment Plan

Executive Summary

Comprehensive **16+ week deployment plan** for ScarCoin mainnet launch, emphasizing **security-first approach** and operational excellence.

🛡️ Security-First Approach

Mandatory audits by 2+ independent firms, comprehensive vulnerability assessment, and zero-tolerance for critical issues.

📚 Phased Deployment Strategy

Three-phase rollout: Limited Launch → Gradual Rollout → Full Launch with conservative parameters and monitoring.
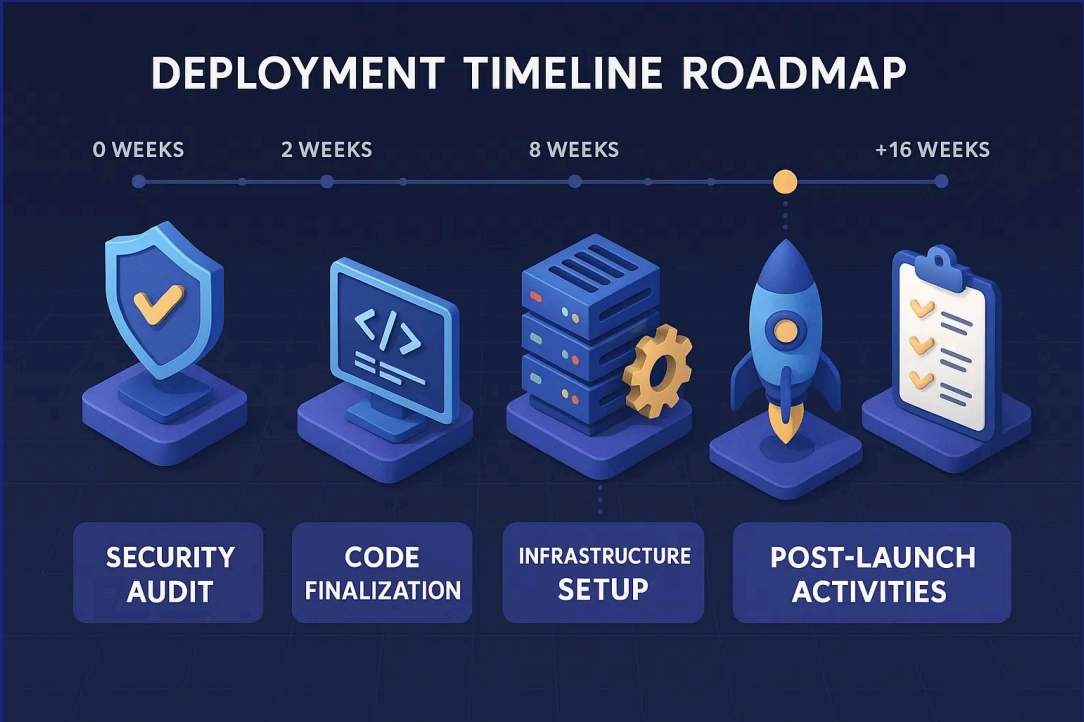
📈 Risk Management Framework

Comprehensive assessment of technical, economic, and operational risks with detailed mitigation strategies.

👥 Multi-Signature Governance

3-of-5 multisig wallet for contract ownership with hardware security modules and geographic distribution.

👁 24/7 Monitoring & Operations

Real-time monitoring, automated alerting, and emergency response procedures for continuous operation.

DEPLOYMENT TIMELINE ROADMAP

0 WEEKS 2 WEEKS 8 WEEKS +16 WEEKS

| SECURITY AUDIT | CODE FINALIZATION | INFRASTRUCTURE SETUP | POST-LAUNCH ACTIVITIES |

Security Audit Requirements

Mandatory Security Audits

- ✓ **Primary audit firm** review
- ✓ **Secondary audit firm** validation
- ✓ **Economic model analysis**

Audit Scope

- 🛡 ScarCoin & Oracle functionality
- 🛡 Access control mechanisms
- 🛡 Mint threshold & edge cases

Acceptance Criteria

- ⚠ **Zero critical vulnerabilities**
- ⚠ **All high-severity issues resolved**
- ⚠ Medium issues with mitigation plans

Audit Timeline

- **Weeks 1-4**   Initial audit assessment
- **Weeks 5-6**   Remediation & updates
- **Weeks 7-8**   Final audit & publication

BLOCKCHAIN SECURITY AUDIT

Phased Deployment Strategy

Phase 1: Limited Launch `2 weeks`

- ✓ Deploy contracts with **restricted minting** capabilities
- ✓ Whitelist-only access for controlled testing
- ✓ Continuous monitoring of system behavior and oracle performance

Phase 2: Gradual Rollout `4 weeks`

- ✓ Incrementally **increase minting limits** based on performance
- ✓ Expand user access beyond initial whitelist
- ✓ Monitor oracle updates and system stability

Phase 3: Full Launch `Ongoing`

- ✓ Remove artificial restrictions and open **public access**
- ✓ Enable full feature availability and integrations
- ✓ Maintain enhanced monitoring and support

Conservative Launch Parameters

- ⚙ Initial mint threshold: **0.700000** (higher than testnet)
- ⚙ Extended cooldown period: **60 minutes** (vs. 30 min on testnet)

Phase 1
Limited Launch → Phase 2
Gradual Rollout → Phase 3
Full Launch
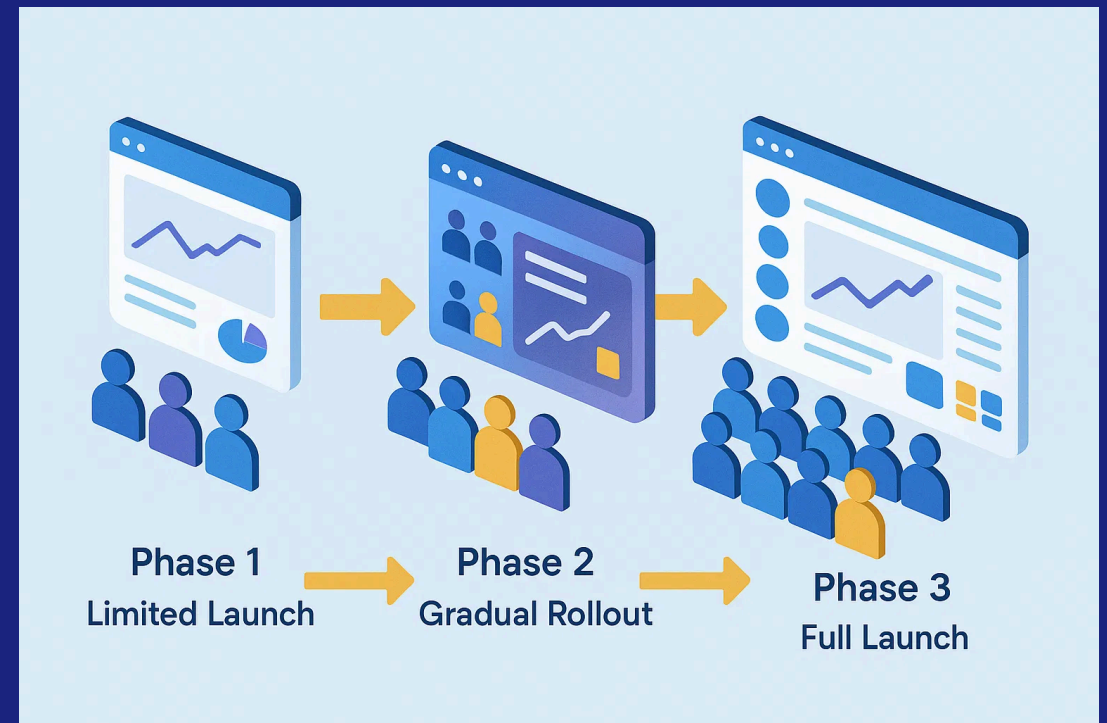
Risk Management Framework

</> Technical Risks

Smart Contract: Comprehensive audits, bug bounty program, emergency pause functionality

Oracle Failure: Multiple data sources, validation checks, manual override capabilities

Network Issues: Gas optimization, transaction monitoring, alternative deployment options

📈 Economic Risks

Market Volatility: Conservative threshold settings, gradual parameter adjustments

Liquidity Issues: Phased launch approach, market maker partnerships

Token Economics: Economic model analysis, parameter optimization

👥 Operational Risks

Team Availability: Cross-training, documented procedures, emergency response team

Infrastructure: Multiple provider redundancy, backup deployment systems

Security Breach: Hardware security modules, access logging, key rotation

Emergency Response Protocol

Immediate containment (0-1h), short-term mitigation (1-24h), and long-term resolution (1-7d) procedures for critical incidents with clear escalation paths and communication templates.

Monitoring & Operations

📈 Real-Time Monitoring

- ⏱ Transaction volume and patterns tracking
- 🪙 Minting activity and success rates
- 🗄 Oracle index updates and anomalies
- ⛽ Gas usage and optimization opportunities

🔔 Alerting System

- ❗ Critical Alerts: Security incidents, oracle anomalies, system downtime
- ⚠️ Warning Alerts: High gas usage, unusual volumes, performance degradation
- ℹ️ Info Alerts: Regular status updates, milestone achievements

🛡 Security Monitoring

- 🔓 Unusual transaction pattern detection
- ⇄ Large token movement tracking
- 🔑 Access control violation alerts

✅ Operational Procedures

- 📅 **Daily:** System health checks, transaction logs, oracle data verification
- 📅 **Weekly:** Performance analysis, security review, stakeholder reporting
- 📅 **Monthly:** Full system audit, parameter optimization, disaster recovery testing

BLOCKCHAIN MONITORING

REAL-TIME MONITORING

PERFORMANCE METRICS

SECURITY MONITORING

OPERATIONS

Timeline & Milestones

Weeks 1-4 **Security Audit**
- ✓ Contract submission to auditors
- ✓ Initial vulnerability assessment

Weeks 5-8 **Code Finalization**
- ✓ Implement audit recommendations
- ✓ Multi-signature wallet setup

Weeks 9-10 **Deployment**
- ✓ Final audit completion
- ✓ **Mainnet deployment**

Weeks 11-16+ **Phased Rollout**
- ✓ Phase 1: Limited access (Week 11)
- ✓ Phase 3: Full operations (Week 16+)

DEPLOYMENT TIMELINE ROADMAP

0 WEEKS   2 WEEKS   8 WEEKS   +16 WEEKS

SECURITY AUDIT   **CODE FINALIZATION**   **INFRASTRUCTURE SETUP**   **POST-LAUNCH ACTIVITIES**

Team Responsibilities

🖥 Technical Lead

› Oversee contract deployment and technical implementation

› Coordinate with audit firms and implement recommendations

› Manage multi-signature setup and technical documentation

🛡 Security Officer

› Implement security protocols and access controls

› Monitor for security threats and vulnerabilities

› Lead incident response for security-related events

## 📈 Operations Manager

› Establish monitoring systems and operational procedures

› Coordinate phased deployment activities

› Manage day-to-day operations and performance optimization

👥 Community Manager

› Coordinate communications during deployment phases

› Manage user onboarding for phased rollout

› Collect and relay community feedback to technical team

External Partners

🗣 **Audit Firms:** Security assessment and recommendations

🗣 **Legal Advisors:** Regulatory compliance guidance

🗣 **Infrastructure Providers:** Node operation and RPC services

TEAM ROLES AND RESPONSIBILITIES FOR BLOCKCHAIN DEPLOYMENT
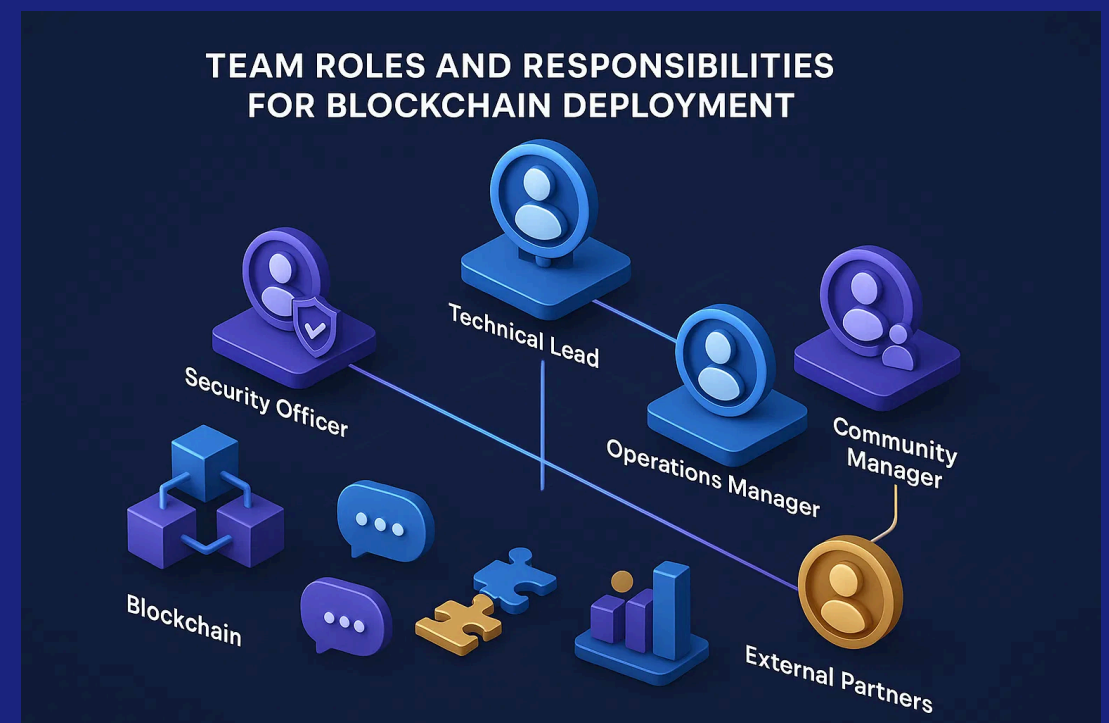
Technical Implementation

🔑 Multi-Signature Wallet Setup

- ✅ **3-of-5 configuration** requiring multiple approvers for critical operations
- ✅ Hardware security modules (HSMs) for all key holders
- ✅ Geographic distribution of key holders across multiple time zones

🗄 Oracle Management System

- ✅ **Enhanced data validation** with multiple source verification
- ✅ Tiered update frequency based on market volatility
- ✅ Anomaly detection with automatic alerts and manual override

</> Enhanced Deployment Scripts

- ✅ Comprehensive pre-deployment validation checks
- ✅ Automated post-deployment verification
- ✅ Detailed logging and transaction monitoring

```
// Example validation check
if (!deploymentInfo.owner ||
!ethers.utils.isAddress(deploymentInfo.owner)) {
  throw new Error("Invalid owner address");
}
```

🛡 Contract Verification & Monitoring

- ✅ Automated Polygonscan verification with source code publishing
- ✅ Real-time transaction monitoring with Tenderly integration
- ✅ Comprehensive health check system with 8+ validation tests

BLOCKCHAIN DEPLOYMENT

SMART CONTRACT

SMART CONTRACT

Success Criteria & Next Steps

✅ Key Success Metrics

📈 **Technical Performance:** Zero critical errors, 100% uptime, gas optimization within 10% of estimates

⇄ **Transaction Processing:** Successful minting operations, proper oracle integration

🛡 **Security Validation:** All contracts verified, ownership confirmed, access controls validated

✅ Validation Checklist

📋 Contract verification on Polygonscan complete

📋 Multi-signature wallet ownership confirmed

📋 Oracle index updates functioning correctly

📋 Mint threshold parameters properly configured

💓 Ongoing Monitoring

📊 Daily health checks and performance metrics

👤 Weekly security reviews and vulnerability scanning

🔄 Monthly parameter optimization based on performance data

Future Roadmap

➡ Enhanced oracle data sources and redundancy

➡ Community governance integration

➡ Cross-chain expansion and interoperability